Periodic Binary Sequences With Very Good Autocorrelation Properties

S. Tyler and J. Loftsson
Reliability Engineering Section

Computer searches were performed using both an 8086 microprocessor and a Cyber 750 mainframe to find repeated binary phase coded waveforms with very good matched and mismatched autocorrelation properties. The best results for every period up to 64 are given. Sequences with optimal peak sidelobes were discovered for each of these periods. These sequences have extensive applications in radar and communications, particularly in situations when there are very unfavorable signal-to-noise ratios. The best sequence of period 64 when processed using a mismatched filter giving no sidelobes has a reduction in the main lobe of less than 0.23 dB.

I. Introduction

This article discusses binary sequences of lengths up to 64 with very good periodic autocorrelations.

For short periods, optimal sequences are well known. For sequence periods which equal 3 mod 4 and are prime or are of the form $2^n - 1$, direct methods for finding a sequence with, in some respects, an optimal autocorrelation are also known.

However, for most other periods, there is no known practical algorithm for deriving optimal sequences (Ref. 1). Since sequences of lengths of about 40 or more are too long to be subjected to an exhaustive computer search (Ref. 2), smaller searches must be made and practical algorithms developed which may lead to sequences with good periodic autocorrelations.

Periodic sequences such as these have great practical value in radar (Ref. 3) and communications (Ref. 4), especially in situations with extremely adverse signal-to-noise ratios. These sequences also have value in artificial intelligence, since analog solutions and pattern recognition methods may apply and be far superior to digital methods for finding excellent (but not the best) long sequences. In addition, such sequences can be used in cryptography to provide derivable "code books" in situations where "two-key" encryption is not desired. Finally, optimal sequences are ideal for use in searches for extraterrestial intelligence; not only are they easy to detect, they also advertise current levels of technology.

One example of the use of such a sequence is the Venus ranging experiment by MIT's Lincoln Laboratory in 1959 and 1961. A binary "pseudorandom" shift register sequence of period $2^{13} - 1 = 8191$ was used to determine whether to

transmit "pulse" or "no pulse" in consecutive time intervals. This sequence was very easy to synthesize and had the property that its autocorrelation was recoverable despite a noise-to-signal excess of many decibels (Ref. 5). Due to the difficulty of analyzing an 8191-sequence, planetary ranging sequence periods are generally shorter ($2^8 - 1 = 255$ is common) or at least factorable. Even a sequence of period 8192 would be relatively easy to analyze using Fast Fourier Transform techniques. However, the computers of 1959 did not have sufficient capability to permit the synthesis of an adequate 8192-sequence. The ranging systems used at the Jet Propulsion Laboratory have tended to use Boolean combinations of several shorter sequences to facilitate rapid acquisition. The combination is used to specify phase modulation on a continuous wave carrier; this technique requires less maximum power output than does amplitude modulation (Ref. 5).

This article will discuss the following:

Matched Periodic Binary Sequences: A description and example of how to evaluate the autocorrelation of a sequence.

Calculation of Mismatched Values: A description and example of how to calculate the main-lobe loss when a sequence is analyzed by a mismatched filter.

Sequence Generation Techniques: A description of the techniques used to generate good sequences.

Results: Tables of the best values found, both for matched values and those analyzed by a mismatched filter.

Some of the values listed are "optimal"; others are merely the best the authors have been able to obtain to date. The main emphasis has been on finding the best value for a sequence of period 64 – the best previous value was improved by over 27%.

II. Matched Periodic Binary Sequences

A binary sequence (or binary code) is a string of bits. It can be thought of as a vector, **c**, where each c_i is a plus one or a minus one. A periodic sequence is one which is continuously repeated; for a binary sequence of period j, $c_{\ell+j} = c_\ell$ for all ℓ.

The "autocorrelation," **a**, of a sequence **c** of period j *is:*

$$a_\ell = \sum_{k=1}^{j} c_k \, c_{k+\ell}$$

When the autocorrelation is normalized by dividing it by j, it is called the "autocorrelation function."

Here a_j, a_{2j}, a_{3j} and so forth are "main lobes," the remaining a_ℓ are "sidelobes;" a is considered to have j "elements," one main lobe and $j - 1$ sidelobes.

For a sequence to have "good matched autocorrelation properties," it must satisfy at least one of the following criteria:

(1) The peak sidelobe in the autocorrelation is small.

(2) The sum of the squares of the sidelobes in the autocorrelation is small.

These concepts are illustrated by means of an example. Consider a sequence of period 7:

$$- - + + - + +$$

To get the elements of the autocorrelation, suppose the following:

$- - + + - + +$	is the original sequence. Then
$+ - - + + - +$	is the sequence shifted one position.
$- + - + - - +$	is the arithmetic product for each position. The number –1 is the sum of these products; it is the first sidelobe element of the autocorrelation.

Shifting by 2,

$$\begin{array}{l} - - + + - + + \\ + + - - + + - \\ \hline - - - - - + - \end{array}$$

The number –5 is the sum; it is the second sidelobe element of the autocorrelation.

Shifting by 3,

$$\begin{array}{l} - - + + - + + \\ - + + - - + + \\ \hline + - + - + + + \end{array}$$

The number +3 is the sum and the next element in the autocorrelation.

Shifts by 4, 5, and 6 positions are equivalent to those of 3, 2, and 1. The last element of the autocorrelation is the main lobe. It corresponds to the original unshifted sequence. The other elements are the sidelobes (the main lobe is not a sidelobe). Thus, the autocorrelation of $- - + + - + +$ is -1, -5,

3, 3, –5, –1, 7. Here P = peak sidelobe magnitude = 5; M = sum of squares of sidelobes = 70.

An "optimal" sequence for period 7 is + + + – – + – and has the autocorrelation -1, -1, -1, -1, -1, -1, 7, where $P = 1$, $M = 6$.

For period 8, + + + + – + – – is an optimal sequence. It has the autocorrelation, 0, 0, 0, –4, 0, 0, 0, 8. Here $P = 4$, $M = 16$.

The remainder of this section shows the connection between the autocorrelation and the Fourier transform.

As a preliminary, it should be noted that $a_\ell = a_{j+\ell} = a_{j-\ell}$ for all ℓ:

$$a_{j+\ell} = \sum_{k=1}^{j} c_k\, c_{k+j+\ell} = \sum_{k=1}^{j} c_k\, c_{k+\ell} = a_\ell = \sum_{k=\ell+1}^{\ell+j} c_{k-\ell}\, c_k$$

$$= \sum_{k=1}^{j} c_{k-\ell}\, c_k = \sum_{k=1}^{j} c_{k+j-\ell}\, c_k = a_{j-\ell}$$

It is useful to have a matrix **Z** which satisfies **a** = **Zc**, that is:

$$a_\ell = \sum_{k=1}^{j} Z_{k\ell}\, c_k$$

However, the "circular convolution matrix" **R** of the sequence **c** is actually a $j \times j$ matrix satisfying:

$$a_{\ell-1} = \sum_{k=1}^{j} R_{k\ell}\, c_k$$

Since

$$a_{\ell-1} = a_{\ell+j-1}$$

$$= \sum_{k=1}^{j} c_k\, c_{k+\ell+j-1}$$

$$= \sum_{k=1}^{j} c_k\, c_{k+1-\ell}$$

Then $R_{k\ell} = c_{k+1-\ell}$

The "Fourier transform" λ of the sequence **c**, a vector $\lambda = \mathbf{Dc}$, satisfies:

$$\lambda_\ell = \sum_{k=1}^{j} D_{k\ell}\, c_k$$

where

$$D_{k\ell} = \frac{1}{\sqrt{j}} \omega^{(k-1)(\ell-1)}$$

$$\omega = \exp(2\pi i/j)$$

$$i = \sqrt{-1}$$

The sequence **c** can be restored from λ by means of the "inverse Fourier transform":

$$c_k = \sum_{m=1}^{j} D^*_{km}\, \lambda^*_m$$

where

λ^*_m = the complex conjugate of λ_m

$$D^*_{km} = \frac{1}{\sqrt{j}}\, \omega^{j-(k-1)(m-1)}$$

The circular convolution matrix can therefore be expanded in terms of an inverse Fourier transform:

$$R_{k\ell} = c_{k+1-\ell} = \frac{1}{\sqrt{j}} \sum_{m=1}^{j} \omega^{j-(k+1-\ell-1)(m-1)}\, \lambda^*_m$$

$$= \frac{1}{\sqrt{j}} \sum_{m=1}^{j} \omega^{j-(k-1)(m-1)}\, \lambda^*_m\, \omega^{(m-1)(\ell-1)}$$

$$= \sqrt{j} \sum_{m=1}^{j} D^*_{km}\, \lambda^*_m\, D_{m\ell}$$

Thus

$$\mathbf{R} = \sqrt{j}\, \mathbf{D}^*\, \Lambda^* D$$

where

$$\Lambda^*_{ij} = \delta_{ij} \lambda^*_i$$

$$\delta_{ij} = 1 \qquad i = j$$

$$= 0 \qquad i \neq j$$

The criterion of optimality that average sidelobe response be minimized with respect to mainlobe response means minimizing

$$\frac{\sum_{\varrho=1}^{j-1} a_\varrho^2}{a_j^2}$$

Since a_j^2 always equals j^2, this is equivalent to minimizing the sum of the squares of the sidelobes:

$$\sum_{\varrho=1}^{j-1} a_\varrho^2$$

which is in turn the same as minimizing

$$\sum_{\varrho=1}^{j} a_\varrho^2 = \sum_{\varrho=1}^{j} a^*_{1-\varrho}\, a_{1-\varrho} = \boldsymbol{c}^* \boldsymbol{R}^* \boldsymbol{R}\boldsymbol{c}$$

$$= j\, \boldsymbol{c}^* \boldsymbol{D}^* |\Lambda|^2 \boldsymbol{D}\boldsymbol{c} = j \lambda^* |\Lambda|^2 \lambda$$

This is equivalent to minimizing

$$\sum_{i=1}^{j} |\lambda_i|^4$$

If evaluations of sequences are to be performed on a computer in a language which includes no bit manipulation instructions, calculating the λ_i is more efficient than evaluating the autocorrelation. It requires j^2 multiplications to calculate the autocorrelation unless individual bits are used. To calculate the λ_i using a Fast Fourier Transform (FFT) requires fewer than j ($\log_2 j$) operations. However, in an assembly language, a maximum of j + (j/word length) logical operations are needed to replace the j^2 multiplications in calculating the autocorrelation.

When calculating the λ_i, it is helpful to check the normalization. By Parseval's theorem:

$$\sum_{i=1}^{j} |\lambda_i|^2 = j$$

III. Calculation of Mismatched Values

When sequences are used for ranging, they are phase coded rather than amplitude modulated. Since the signal-to-noise ratio (SNR) is expected to be very low, it is generally favorable to use maximum amplitude throughout; amplitude modulation would be inconsistent with this requirement. Similarly, periodic rather than aperiodic waveforms are generally used to increase the redundancy of the information. A single (aperiodic) sequence uses 2^n bits to transmit only n bits of information (the information transmitted being the displacement of the starting point of the sequence). However, a repeating (periodic) sequence uses $m \times 2^n$ bits to transmit the same information, where m is the number of repetitions which are processed (whether this is truly the best way to use $m \times 2^n$ bits to transmit n bits of information in a noisy environment is not the issue).

The detection procedure is equivalent to comparing the incoming signal to a template consisting of the original sequence and "moving the template around" until it matches the signal. In this situation the autocorrelation is the "output" of a "matched receiver."

Although the signal is not amplitude modulated, the *template* may have some amplitude modulation. The incoming sequence is then no longer correlated with itself but with a similar "weighted" sequence. By varying the amplitude of each bit in this template sequence, the sidelobes can be reduced or even eliminated.

The cross-correlation, $\boldsymbol{x}$, of two sequences, $\boldsymbol{b}$ and $\boldsymbol{c}$, each of period j is:

$$x_\varrho = \sum_{k=1}^{j} b_k\, c_{k+\varrho}$$

Let $\boldsymbol{c}$ be a binary sequence which is to be cross-correlated with $\boldsymbol{b}$, a sequence composed of real numbers. Sequences $\boldsymbol{c}$ and $\boldsymbol{b}$ are related by the "weighting function" $\boldsymbol{t}$.

$$b_\varrho = t_\varrho\, c_\varrho$$

In this case, $\boldsymbol{b}$ can be considered a "mismatched filter" to $\boldsymbol{c}$. To normalize this mismatched filter:

$$\sum_{\varrho=1}^{j} b_\varrho^2 = j$$

For a given sequence, there usually exists a mismatched filter which can be used to mathematically operate on the sequence so as to reduce all the sidelobe values to zero. For $j > 4$, however, the main-lobe value will also be reduced somewhat. The sequence with the best mismatched (cross-correlation) properties is the one which has the smallest decrease in main-lobe value and therefore has the smallest ratio of

$$\frac{j^2}{x_j} = \frac{j^2}{\sum_{k=1}^{j} b_k c_k}$$

The circular convolution matrix $\boldsymbol{R}$ for mismatched sequences is:

$$x_{\ell-1} = \sum_{k=1}^{j} R_{k\ell} b_k$$

The average-to-peak cross-correlation response is then:

$$\frac{\sum_{i=1}^{j-1} x_i^2}{x_j^2}$$

Minimizing the above is equivalent to minimizing

$$\frac{\boldsymbol{b}^* \boldsymbol{R}\boldsymbol{R}^* \boldsymbol{b}}{\boldsymbol{b}^* \boldsymbol{c}\boldsymbol{c}^* \boldsymbol{b}}$$

which occurs when

$$\boldsymbol{b} = (\boldsymbol{R}\boldsymbol{R}^*)^{-1} \boldsymbol{c}$$

$$= \frac{1}{j} \boldsymbol{D}^* \, | \Lambda |^{-2} \, \boldsymbol{D}\boldsymbol{c}$$

This gives b_k proportional to

$$\sum_{\ell=1}^{j} D_{k\ell}^* \frac{1}{\lambda_\ell^*}$$

This choice of $\boldsymbol{b}$ zeros the cross-correlation (Ref. 5). The only constraint necessary for $\boldsymbol{R}\boldsymbol{R}^*$ to be nonsingular is that $|\lambda_i|^2 > 0$ for all i, i.e., that $\boldsymbol{R}\boldsymbol{R}^*$ is positive definite.

The best sequence is the one which minimizes

$$\frac{1}{j} \sum_{i=1}^{j} \frac{1}{|\lambda_i|^2}$$

In practice, smaller weights may be chosen to reduce the SNR loss. In this case, the sidelobes will be reduced rather than eliminated.

The following example illustrates the derivation of the appropriate mismatched filter for a given sequence.

Consider the sequence of length 8 discussed in the previous section. The sequence + + + + - + - - has autocorrelation 0, 0, 0, -4, 0, 0, 0, 8, where P = peak sidelobe magnitude = 4 and where M = sum of squares of sidelobes = 16.

It will be shown that when this sequence is analyzed with the appropriate mismatched filter, $\boldsymbol{b}$, the cross-correlation becomes $(0, 0, 0, 0, 0, 0, 0, x_j)$.

The elements of the mismatched filter, b_k, must be normalized so that

$$\frac{1}{8} \sum_{k=1}^{8} b_k^2 = 1$$

in order to obtain the correct value of x_j.

The loss in SNR for the mismatched filter is then $L = (8/x_j)^2$. To actually calculate L:

$$L = \frac{1}{8} \sum_{k=1}^{8} \frac{1}{|\lambda_i|^2}$$

where the λ_k are elements of the Fourier transform of the original sequence:

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix} = \sqrt{\frac{1}{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega & i & -\omega^* & -1 & -\omega & -i & \omega^* \\ 1 & i & -1 & -i & 1 & i & -1 & -i \\ 1 & -\omega^* & -i & \omega & -1 & \omega^* & i & -\omega \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -\omega & i & \omega^* & -1 & \omega & -i & -\omega^* \\ 1 & -i & -1 & i & 1 & -i & -1 & i \\ 1 & \omega^* & -i & -\omega & -1 & -\omega^* & i & \omega \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

where

$$i = \sqrt{-1} = \exp(i\pi/2)$$

$$\omega = \sqrt[4]{-1} = \exp(i\pi/4) = (1 + i)/\sqrt{2}$$

$$\lambda = \frac{1}{\sqrt{8}} \begin{bmatrix} 2 \\ 2(1+i) - \sqrt{2}(1-i) \\ 2i \\ 2(1-i) + \sqrt{2}(1+i) \\ -2 \\ 2(1+i) + \sqrt{2}(1-i) \\ -2i \\ 2(1-i) - \sqrt{2}(1+i) \end{bmatrix}$$

Thus

$$|\lambda|^2 = \frac{1}{8}(4, 12, 4, 12, 4, 12, 4, 12)$$

$$\frac{1}{|\lambda|^2} = (2, 2/3, 2, 2/3, 2, 2/3, 2, 2/3)$$

$$L = \frac{1}{8}\sum_{k=1}^{8} \frac{1}{|\lambda_k|^2}$$

$$= \frac{1}{8}(8 + 8/3)$$

$$= 4/3$$

$$= 1.3333$$

The loss in dB is $10 \log_{10}(1.3333) = 1.250$ dB.

For this particular case, the mismatched filter elements can be found by inspection. They also can be calculated as follows:

$$b_i \propto \sum_{k=1}^{j} D_{ik}^{*} \frac{\lambda_k}{|\lambda_k|^2}$$

where D_{ik}^{*} is the inverse Fourier transform.

There is no need to normalize the b_i at this point; it can always be done later since

$$\sum_{i=1}^{j} b_i^2 = j$$

The filter **b** is proportional to

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega^* & -i & -\omega & -1 & -\omega^* & i & \omega \\ 1 & -i & -1 & i & 1 & -i & -1 & i \\ 1 & -\omega & i & \omega^* & -1 & \omega & -i & -\omega^* \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -\omega^* & -i & \omega & -1 & \omega^* & i & -\omega \\ 1 & i & -1 & -i & 1 & i & -1 & -i \\ 1 & \omega & i & -\omega^* & -1 & -\omega & -i & \omega^* \end{bmatrix} \begin{bmatrix} 6 \\ 2(1+i) - \sqrt{2}(1-i) \\ 6i \\ 2(1-i) + \sqrt{2}(1+i) \\ -6 \\ 2(1+i) + \sqrt{2}(1-i) \\ -6i \\ 2(1-i) - \sqrt{2}(1+i) \end{bmatrix} = 8 \begin{bmatrix} 1 \\ 3 \\ 1 \\ 1 \\ -1 \\ 3 \\ -1 \\ -1 \end{bmatrix}$$

With **b** properly normalized:

$$\mathbf{b} = \frac{1}{\sqrt{3}}(1, 3, 1, 1, -1, 3, -1, -1)$$

It is easily verified that the cross-correlation of

$$(1, 1, 1, 1, -1, 1, -1, -1)$$

and

$$\frac{1}{\sqrt{3}}(1, 3, 1, 1, -1, 3, -1, -1)$$

is

$$(0, 0, 0, 0, 0, 0, 0, 4\sqrt{3})$$

IV. Sequence Generation Techniques

A. Iterative Improvement

The basic concept in deriving a good sequence is to start with an arbitrary sequence of period j and perturb one bit of the sequence to create a new sequence. Then the criteria of optimality developed in the previous section are used to decide whether the new sequence is superior to the old one. If the new sequence is inferior, a new bit of the old sequence is perturbed; if the new sequence is superior, it replaces the old sequence and is itself then modified by one bit. Eventually a "locally optimal" sequence is thus obtained. A new initial sequence is chosen, and the procedure is repeated as often as desired.

Perturbing a sequence by 2 bits or 3 bits was also tried; it yielded significantly inferior results to 1-bit modifications.

It may appear that one should look at, say, all ${}_jC_4$ sequences which vary by four bits from the best sequence

obtained by iterative improvement once one-bit, two-bit, and three-bit modifications have failed to improve it rather than abandon the sequence and start over. Certainly, it can not hurt to apply this technique to the best sequence ever found. However, in general, the time spent in an often futile attempt (via 4-bit modifications) to improve one already "locally optimized" sequence could be better spent by locally optimizing a thousand new sequences from scratch.

The iterative procedure also produces good sequences faster than does an exhaustive search. For $j = 43$, at least one out of every 10^7 sequences examined had $M = 42$; for an exhaustive search, only one out of every 10^{10} sequences would have that value. Even if the exhaustive search examined sequences only when

$$\sum_{i=1}^{j} c_i = \pm 1$$

only one of 10^9 sequences would have $M = 42$. An actual exhaustive search might be restricted to sequences beginning with two or more +1's followed by a -1; this would not increase the rate of finding $M = 42$ sequences.

The best results were obtained by minimizing M, but good results should also be obtainable by minimizing L. When iteratively improving a sequence by calculating L, one need not continually recalculate the sum in

$$\lambda_\ell = \sum_{k=1}^{j} D_{k\ell}\, c_k$$

If element c_n of the original sequence is to be perturbed, the new Fourier transform elements, λ'_ℓ, are always simply:

$$\lambda'_\ell = \lambda_\ell \pm 2D_{n\ell} = \lambda_\ell \pm (2/\sqrt{j}) \exp(2\pi i(n-1)(\ell - i)/j)$$

which is noticeably more efficient, especially when j is prime so that an FFT does not help.

It is also possible to modify a sequence iteratively by noting which elements of the Fourier transform are farthest from unity and then perturbing the appropriate bit or bits in the sequence to improve these worst values of λ_i.

B. Choosing an Initial Sequence

Since numerous sequences were to be chosen, an important criterion was to avoid accidentally repeating initial sequences. This was done by making each 4-bit "nibble" of the first initial sequence different and then systematically changing the sequence of nibbles. Rotations of initial sequences are unlikely to give identical results after iteration (but ones complementation will). Thus for $j = 64$, this procedure can supply $16!/2 > 10^{13}$ initial sequences while for $j = 48$ it gives $16!/(2 \times 4!) > 4 \times 10^{11}$ initial sequences.

Attempts to improve on the choice of initial sequences by modifying sequences of periods $j \pm 1$, $j \pm 4$, $2j$, $j/2$, $j/4$, $\sqrt{j}$, $j_1 + j_2 = j$, and so forth (where the unmodified sequence had good autocorrelation properties) did not yield better results. It is also difficult to synthesize a large number of such initial sequences.

It may seem that a minor modification of a sequence with, say, $j = 1023$ and $M = 1022$ will give a good sequence with $j = 1024$; this is simply not true. An even worse idea would be to create a $j = 1023$ sequence with a shift register and then pretend it has $j = 1024$ and analyze it with a Fast Fourier transform.

C. Methods of Finding Good Sequences by Inspection

1. Quadratic residue sequences. For odd prime periods, j, one forms a quadratic residue sequence by setting to -1 all elements c_i for which $i = n^2 \bmod j$ for some integer $n < j/2$. The remaining elements are set to +1. These sequences give $M = j - 1$ (always optimal) for $j = 3 \bmod 4$. For $j = 1 \bmod 4$, half of the a_i equal +1 and the remaining a_i are -3, thus $M = 5(j - 1)$ which presumably is never optimal, merely good.

For example let $j = 11$. Elements 1, 4, 9, $5 = 16 \bmod 11$, and $3 = 25 \bmod 11$ are set to -1. The remaining elements are +1. This gives a sequence with $M = 10$.

An integer I is a quadratic residue modulo n if $m^2 = I$ (modulo n) has a solution for some integer m and $(I, n) = 1$. When p is an odd prime, the Legendre symbol (I/p) is defined as:

$$\left(\frac{I}{p}\right) = \begin{cases} 1 \text{ if } I \text{ is a quadratic residue modulo } p \\ -1 \text{ otherwise} \end{cases}$$

For this reason, quadratic residue sequences are also referred to as Legendre sequences.

Quadratic residue sequences always have sums

$$\sum_{i=1}^{j} c_i = \pm 1$$

As will be shown later, such sequences do not have good mismatched autocorrelation properties. It would be more useful to discover an algorithm (if one exists) to produce the

$P = 3$ sequences with sums of ±7 that dominate Table 2 for $j = 1 \bmod 4$ rather than use quadratic residue sequences with $P = 3$ and sums of ±1.

2. Double prime sequences. For periods, j, which are the products of two odd primes: $j = j_1 j_2$ where $j_2 > j_1$, one can synthesize "double prime" sequences. These sequences are called "twin prime" when $j_2 = j_1 + 2$. Twin prime sequences are always optimal, with $M = j - 1$. Double prime sequences are based on the Jacobi symbol $[I/j]$ where $j = j_1 j_2$ and

$$\left[\frac{I}{j}\right] = \left(\frac{I}{j_1}\right)\left(\frac{I}{j_2}\right)$$

For double primes, one sets to -1 all elements c_i for which $(i, j) = 1$ and $[i/j] = -1$ as well as those for which $(i, j) \neq 1$ and $i = 0 \bmod j_2$. The remaining elements are set to +1. For example, let $j = 35$. For $(i, j) \neq 1$ and $i = 0 \bmod 7$, elements 7, 14, 21, 28, and 35 are set to -1. For $[i/j] = -1$ elements 2, 6, 8, 18, 19, 22, 23, 24, 26, 31, 32, and 34 are set to -1. The rest of the elements are +1. This gives $M = 34$.

For $j_2 = j_1 + 4$, this method gives sequences with autocorrelation element values exclusively of 1 and -3, which are presumably good but not optimal. For $j_2 = j_1 + 6$, the method gives autocorrelation element values exclusively of -1, +3, and -5, which are not necessarily even good.

Triple prime sequences can also be generated using the Jacobi symbol. The first one with three odd primes has $j = 105$, a period which is not investigated in this paper. In addition, it is not manifest that such a sequence should be good, let alone optimal.

3. Shift register sequences. For $j = 2^n - 1$, a "shift register" sequence can always be created with $M = j - 1$. For example let $j = 15 = 2^4 - 1$. Then one must find an irreducible polynomial of order 4 such as:

$$r^4 + r^3 + 1 = 0$$

with a recursion of

$$r_k = r_{k-3} + r_{k-4}$$

where addition is mod 2. Applying this recursion to 0011 gives 001101011110001 which, when one replaces the 0's with -1's, has $M = 14$.

These sequences are produced by "shift registers" which are devices of n consecutive binary storage positions which shift the contents of each position to the next position down the line at regular intervals. To compute the new first position, a mod 2 sum of the contents of some of the previous n positions ($n = 4$ in the above example) is used.

For $j \leqslant 127$, a recursion relation for $j = 2^n - 1$ can always be found of the form:

$$r_k = r_{k-\ell} + r_{k-n}$$

where $0 < \ell < n$, which gives a sequence with $M = j - 1$.

4. Multiplication of sequences. When one multiplies two sequences of relatively prime periods bit by bit, their autocorrelations are also multiplied, element by element. Thus if sequence **c**, with autocorrelation **a**, is formed by multiplying the sequences **d** and **e**, with autocorrelations **u** and **v** respectively, then

$$c_i = d_i e_i$$

and

$$a_i = u_i v_i$$

If **e** is optimal ($P = 1$) with some odd prime period $j_e = 3 \bmod 4$ and **d** is optimal ($P = 0$) of period 4, then **c** has period $j_c = 4j_e$ and $M_c = 16 (j_e - 1) = 4j_c - 16$. For example if **d** is + + + -, and **e** is + + + - - + -, then write **d** as

(+ + + - + + + - + + + - + + + - + + + - + + + - + + + -)

and **e** as

(+ + + - - + - + + + - - + - + + + - - + - + + + - - + -)

Then **c** is

(+ + + + - + - - + + - + + - + - + - - - - + + - - - + +)

To get the autocorrelation:

u = (0 0 0 4 0 0 0 4 0 0 0 4 0 0 0 4 0 0 0 4 0 0 0 4 0 0 0 4)

v = (-1 -1 -1 -1 -1 -1 7 -1 -1 -1 -1 -1 -1 7 -1 -1 -1 -1 -1
-1 7 -1 -1 -1 -1 -1 7)

So

a = (0 0 0 -4 0 0 0 -4 0 0 0 -4 0 0 0 -4
0 0 0 -4 0 0 0 -4 0 0 0 28)

An $M = 192$ sequence for $j = 52$ can be obtained by multiplying the 4 and 13 sequences. The 4-sequence, **d**, is: $d_1 = 1$, $d_2 = 1$, $d_3 = 1$, $d_4 = -1$, and $d_i = d_{i-4}$. In hex notation, replacing the -1 by a 0, this is E. Repeating this thirteen times, **d** = EEEEEEEEEEEEE. The 13-sequence, **y**, is, in hex notation: 1F35. Repeating this four times, **y** = F9AFCD7E6BF35. Then $z_i = d_i y_i$ gives **z** = E8BEDC6F7AE24 which has $M = 192$ as well as an L equal to that of the 13-sequence, namely 1.040. The "exclusive nor" (the ones complement of the "exclusive or") operation is used to perform this multiplication in hex notation.

5. Golay sequences. There is another method to obtain an $M = 192$ sequence for $j = 52$.

Let **y** = F9AFCD7E6BF35, as before. Then let

$$d_i = 1 \qquad 1 \leqslant i \leqslant 26$$

$$d_i = -1 \qquad 27 \leqslant i \leqslant 39$$

$$d_i = 1 \qquad 40 \leqslant i \leqslant 52$$

In hex, **d** = FFFFFFC001FFF

$$z_i = d_i y_i \text{ now gives } z = \text{F9AFCD4195F35}$$

which is a sequence of the form **S S S S** where each **S** is a sequence of length 13. The sequence z has $L = 1.050$. Sequences of the more general form $\mathbf{S_1 S_2 S_1 S_2}$ are called Golay sequences. Sequences of the form **S S S S** do not generally have the property of producing good autocorrelations. However, they do have the property that, when **S** is of length ℓ, the first $\ell = j/4$ sidelobes on either side of the main lobe are 0.

D. Demonstration of Equivalence

Two sequences which are *equivalent* always have the same M value (the converse is not true). Two sequences, **c** and **d**, can be shown to be equivalent, **c** ~ **d**, as follows:

(1) **c** ~ **c**.

(2) If **c** ~ **e** and **e** ~ **d**, then **c** ~ **d**.

(3) If **c** ~ **d**, **d** ~ **c**.

(4) If $c_i = -d_i$ for all i then **c** ~ **d**.

(5) If $c_i = d_{i+k}$ for all i and some k, then **c** ~ **d**.

(6) If $c_i = d_{i \times k}$ for all i and some k where $(i, k) = 1$, then **c** ~ **d**.

As a special case, when $k = j - 1$, $c_i = d_{j-i}$, a "mirror image" sequence.

When an exhaustive search is made, only one sequence from each equivalence class need be examined: an efficient algorithm for performing such a search is not known.

One method for restricting a search to a small number of members of each equivalence class is to examine only sequences beginning with several +1's followed by a -1; it is not evident that any significant advantage can be obtained in this manner, however.

E. Proof of Optimality

All of the best values of P discovered for each j as well as a number of M values are given as optimal. Most of the proofs of optimality are trivial and based only on the fact that when $j = N \bmod 4$, each element of the autocorrelation is also $N \bmod 4$. For example, when $M = j - 1$ for odd j, one may be sure that no smaller value of M can be obtained. Similarly, $M = 4(j - 1)$ must be optimal for $j = 2 \bmod 4$.

Other useful facts for proving optimality are:

(1) No sequence of $j > 4$ has $P = 0$ (Ref. 7).

(2) No sequence of $j > 13$ and $j = 1 \bmod 4$ has $P = 1$.

(3) $$\sum_{i=1}^{j} a_i = \left(\sum_{i=1}^{j} c_i \right)^2$$

This latter equation is easy to derive. Suppose c_i has h 1's and ℓ -1's where $h + \ell = j$. Then

$$\sum_{i=1}^{j} c_i = h - \ell$$

To find the autocorrelation sum, it is sufficient to realize that every element of the sequence will be multiplied by h 1's and ℓ -1's; thus,

$$\sum_{i=1}^{j} a_i = (h - \ell)^2$$

F. Examples of Optimality Proofs

To illustrate methods of optimality proofs, two examples are given

1. Proof for $j = 36$. The period $j = 0 \bmod 4$, so each a_i must be $0 \bmod 4$

$$\sum_{i=1}^{36} a_i = \left(\sum_{i=1}^{36} c_i \right)^2$$

The right hand side of the above equation is an even square (0, 4, 16, 36, 64, 100, . . .). The left hand side is

$$36 + \sum_{i=1}^{35} a_i$$

Thus either

$$\left| \sum_{i=1}^{35} a_i \right| = 0$$

or

$$\left| \sum_{i=1}^{35} a_i \right| \geqslant 20$$

If

$$\left| \sum_{i=1}^{35} a_i \right| \geqslant 20$$

then $M \geqslant 80$. If

$$\left| \sum_{i=1}^{35} a_i \right| = 0$$

then either every $a_i = 0$, which is impossible, or some $|a_i| \geqslant 4$. In this case $a_{j-i} = a_i$ gives two equal deviations from zero (unless $j - i = i$) which can be negated only by two other elements of the autocorrelation. With 4 nonzero elements, $M \geqslant 64$. If $j - i = i$, then this deviation from zero must be negated by two or more elements of the autocorrelation; thus $M \geqslant 8^2 + 4^2 + 4^2 = 96$. So $M = 64$ is optimal.

2. **Proof for $j = 41$.** The period $j = 1 \bmod 4$, so each a_i must be 1 mod 4.

$$\sum_{i=1}^{41} a_i = \left(\sum_{i=1}^{41} c_i \right)^2$$

The right hand side is (1, 9, 25, 49, 81, . . .); therefore

$$41 + \sum_{i=1}^{40} a_i$$

equals the right hand side (RHS). If RHS = 81, 40 a_is = 1 give $M = 40$ and $P = 1$, which is impossible. If 36 a_is = 1, 2 a_is = -3, and 2 a_is = 5, then $M = 36 + 18 + 50 = 104$. If RHS = 49, 32 a_is = 1 and 8 a_is = -3 give $M = 32 + 72 = 104$. If RHS = 121, 30 a_is = 1 and 10 a_is = 5 give $M = 30 + 250 = 280$. These are the minimum deviations from all $a_i = 1$ for the RHSs closest to 81. Thus $M = 104$ is optimal.

G. Further Evidence of Optimality

Evidence of optimality can also be obtained even when an exhaustive search has not been performed and a proof attempt indicates that lower values of M may be possible. When $j = 44$, for example, 2^{44} sequences are possible. However, as our discussion of equivalent sequences has shown, if one sequence has a given M value, so do a number of others:

44 rotations

× 2 +/- interchanges

× 20 modifications by taking every nth element where $(44, n) = 1$.

This appears to give 1760 sequences with the same autocorrelation properties. However, some of these sequences are identical, so the number of equivalent sequences is less than 1760. Nevertheless, for some values of M, more than 2^{10} equivalent sequences with a given M value must exist if any do. So roughly every 10 billionth sequence would have that value. If anywhere near that many sequences were examined iteratively without finding a given M value, it would strongly suggest that either no sequence giving that value existed or that such a sequence could not easily be derived by iterative one-bit modifications of better and better sequences.

When the M-value is underlined in Table 2, the authors feel that no better value exists. When no proof of optimality exists, the best evidence for this is the accumulation of a large number of sequences, many of which are equivalent, of that value. When one hundred sequences of $M = 144$ for $j = 44$ are found but none of $M < 144$ are discovered, there is considerable circumstantial evidence that $M = 144$ is optimal for $j = 44$. On the other hand, $M = 112$ for $j = 60$ may seem very surprising. Prior to the discovery of such a sequence, one might be excused for believing that no such sequence will be found. Yet, when one or two sequences with $M = 112$ are discovered, the evidence against a sequence with $j = 60$ and $M = 80$ or 96 is not overwhelming. Thus, the authors have decided not to underline a value for M in the table unless at least 30 sequences with that M have been discovered independently.

Of course, this criterion is no guarantee of optimality. For example, for $j = 43$, thirty sequences with $M = 138$ were discovered prior to the appearence of a lower value ($M = 42$). The generation of $P = 1$ sequences for $j = 43$ and $j = 47$ by the iterative method in less than 30 minutes of Cyber processing

time also indicates that the M-values listed for $j < 47$ are likely to be optimal.

H. Ratio of Ones to Minus Ones

As can be seen from the preceding section, the best sequences have

$$\left(\sum_{i=1}^{j} c_i\right)^2 \approx j$$

(the DC Fourier element)

$$\Rightarrow \sum_{i=1}^{j} c_i \approx \sqrt{j}$$

for best results. Thus one expects that the best codes of $j \approx 64$ will have

$$\sum_{i=1}^{j} c_i = \pm 8$$

For a period of 64, this gives 36 1's and 28 -1's (or vice versa).

Robert Keston has pointed out that if one splits such a $j = 0 \bmod 4$ sequence, **c**, into two sequences, **f** and **g**, where

$$f_i = c_{2i-1}$$

$$g_i = c_{2i}$$

then either **f** or **g** should have an equal number of 1's and -1's (R. Keston, personal communication, May 1984).

This can provide assistance in selecting initial sequences or discarding unwanted sequences.

If one is looking for a *particular* value of M for a given j, it may be helpful to look at the properties of the a_i and c_i that must be satisfied. For example, at one time the best known M for $j = 48$ was 112. It was hoped that an M of 96 could be obtained. To accomplish this, one must have:

$$\sum_{i=1}^{48} a_i = \left(\sum_{i=1}^{48} c_i\right)^2$$

$$= 16, 36, 64, 100, \ldots$$

$$= \text{RHS}$$

But $|\text{RHS} - 48| > 24$ would mean $M > 96$ so only RHS = 36 or 64 are possible. For RHS to equal 36, there must be an odd number of fours in the autocorrelation; this can not give $M = 96$. Also, RHS = 64 cannot work with 6 fours, since 5 of them, including the middle element would be positive and one (the middle element again, which is impossible) would be negative. So the only sequence which works must have an autocorrelation with +8 in the middle and a +4 on either side so that 48 + 8 + 4 + 4 = 64. The sequence itself must have either 20 or 28 1's. If one takes every other element of the sequence, one will get 8, 12, or 16 1's. These restrictions could make it easier to hunt for such a sequence. Luckily in this case, even without using them, there was ample time to find a sequence with $M = 96$.

I. Sequences With Good Matched but Poor Mismatched Autocorrelation Properties

Sequences with excellent mismatched autocorrelation properties generally have very good matched autocorrelation properties. The converse is not true and is most typically false for quadratic residue, double-prime or shift register sequences. The reason is that for such sequences,

$$\left|\sum_{i=1}^{j} c_i\right| = 1$$

So the first element of the Fourier transform,

$$\lambda_1 = \frac{1}{\sqrt{j}}$$

Suppose $j = 63$. Then

$$\lambda_1 = \frac{1}{\sqrt{63}}$$

$$\frac{1}{\lambda_1^2} = 63$$

Using Parseval's theorem:

$$\sum_{i=1}^{63} \lambda_i^2 = 63$$

Thus

$$\sum_{i=2}^{63} \lambda_i^2 = 63 - \frac{1}{63}$$

At best, all the λ_i for $2 \leqslant i \leqslant 63$ are equal. Then each of these

$$\lambda_i^2 = \frac{1}{62}\left(63 - \frac{1}{63}\right) = \frac{64}{63}$$

So

$$\sum_{i=1}^{63} \frac{1}{|\lambda_i|^2} = 63 + \frac{62 \times 63}{64} = \frac{63^2}{32}$$

Therefore the L value is, at best, 63/32 = 1.96875, which is very poor. The L value of the shift register sequence with $j = 63$ and $M = 62$ in Table 2 actually is 63/32; the remaining Fourier components are equal.

By the above argument the best possible L value for a shift register sequence of $j = 2^n - 1$ is:

$$L = \frac{2^n - 1}{2^{n-1}}$$

Unless n is small, $L \approx 2$, or about 3 dB.

The DC component of the Fourier transform, greatly elevated due to the near equality of 1's and -1's in the sequence, always produces an L value which represents roughly a 3 dB loss in signal; this compares very unfavorably with the 0.15 dB to 0.25 dB losses corresponding to some of the sequences with better ratios of 1's to -1's. Sequences of period $4j$ formed by multiplying an $L = 1$ sequence of period 4 by a shift register sequence are no better, as the L-value of the sequence with period $4j$ equals that of the sequence with period j.

This provides another incentive for not investigating shift register sequences exclusively. Not only is it a nuisance to analyze such sequences; in addition their autocorrelation properties are, in some respects, not very good.

John Bailey has offered a solution to this problem; eliminate the DC component. One method would be to have +1 and -1 be out of phase by other than 180° (Ref. 8).

For a two phase sequence the modification is:

Element of Shift-Register Sequence	Element of Modified Sequence
+1	+1
-1	$-\exp i\beta$

where

$$\beta = \tan^{-1}\left(\frac{2\sqrt{j}}{j-1}\right)$$

For example, if the unmodified sequence is

$$+ + + - - + -$$

then the modified sequence is

$$c = (1, 1, 1, -\exp i\beta, -\exp i\beta, 1, -\exp i\beta)$$

For complex elements, the autocorrelation is:

$$a_\ell = \sum_{k=1}^{j} c_k^* c_{k+\ell}$$

In our example, $a_7 = 7$

$$a_\ell = 3 - 2(\exp i\beta) - 2(\exp -i\beta)$$

$$= 3 - 4\cos\beta$$

for $1 \leqslant \ell \leqslant 6$ where

$$\beta = \tan^{-1}\frac{\sqrt{7}}{3}$$

$$\tan^2\beta + 1 = \frac{1}{\cos^2\beta} \Rightarrow \cos^2\beta = \frac{9}{16}$$

Putting β in the first quadrant, $\cos\beta = 3/4$, so $a_\ell = 0$ for $1 \leqslant \ell \leqslant 6$.

In general, when $j = 2^N - 1$

$$a_\ell = \frac{j-1}{2} - \frac{j+1}{2}\cos\beta$$

for $1 \leqslant \ell < j$. Thus

$$\tan^2\beta = \frac{4j}{(j-1)^2}$$

and

$$\cos^2\beta = \frac{(j-1)^2}{(j+1)^2}$$

which, with β in the first quadrant, gives

$$a_\ell = 0 \qquad (\ell \neq 0 \bmod j)$$

One could also derive a 3-phase sequence as a product of two 2-phase sequences:

Elements of Unmodified Sequences		Element of Modified Product
1	1	1
-1	-1	1
1	-1	$-\exp(-i\beta)$
-1	1	$-\exp i\beta$

Once again,

$$\beta = \tan^{-1}\left(\frac{2\sqrt{j}}{j-1}\right)$$

If one wishes to zero all sidelobes without a decrease in SNR, one can also let $j = 2^{2N}$ and create a sequence with $\sqrt{j}$ phases; a complete discussion of this would be too far afield of the topic of binary sequences.

V. Results

Table 2 shows the best sequences for periods 28 to 64 for both matched and mismatched cases. Table 1, showing the results for periods 3 to 27 (Ref. 9) is included for completeness.

In Table 2, the heading j gives the period (length) of the sequence; P gives the lowest value of the peak sidelobe; M gives the lowest sum of the squares of the sidelobes discovered for any sequence of period j. When the sequence with the optimal peak sidelobe has a higher M, both values are given. When two references are given on the same line, the first one refers to the matched sequence and the second one to the mismatched sequence. A reference of "X" refers to this article.

When the value for P, M or L is in parentheses, the authors feel that a better, but as yet undiscovered, sequence may exist. When the value is underlined, it is unlikely that a better value exists. In all other cases, the value can be proved to be optimal. All values for P are optimal unless two values are given for a specific j, in which case the lower one is optimal.

The sequences are written in hex notation. The first bit is always a plus sign. For example, the sequence for 29 is given in hex as 14A7C111. In binary this would be 0001 0100 1010 0111 1100 0001 0001 0001. By replacing 0 with a minus sign, and 1 with a plus sign, and removing the leading zeros, we get the sequence:

+ - + - - + - + - - + + + + + - - - - - + - - - + - - - +

References

1. Boehmer, A., Binary Pulse Compression Codes, *IEEE Transactions on Information Theory*, Volume 13, No. 2, p. 156, April 1967.

2. Lindner, J., Binary Sequences up to Length 40 with Best Possible Autocorrelation Functions, *Electronic Letters*, Vol. II, No. 21, p. 507, 16 October 1975.

3. MacMullen, A., *Radar Antennas, Transmitters, and Receivers*, pp. PC-1-56, Technology Service Corporation, April 1977.

4. Hoffman de Visme, G., *Binary Sequences*, English Universities Press, 1971.

5. Golomb, S., Baumert, L., Easterling, M., Stiffler, J., and Viterbi, *A., Digital Communications With Space Applications*, Prentice-Hall, 1964.

6. Brennan, L. E., and Reed, I. S., Theory of Adaptive Radars, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-9, March 1973.

7. Turyn, R., *Optimum Codes Study*, Sylvania Electronic Systems Final Report AF19 (604)-5473, 29 January 1960.

8. Bailey, J., *Modified PN Codes With Optimum Autocorrelations*, Technology Service Corporation, August 1978.

9. Tyler, S., and Keston, R., Optimal Periodic Binary Codes of Lengths 28 to 64, *TDA Progress Report 42-57*, March and April 1980, Jet Propulsion Laboratory, Pasadena, CA.

10. Bailey, J., and Tyler, S., *Periodic Binary Waveforms with Optimum Autocorrelation Functions*, Technology Service Corporation, September 1978.

11. Tausworthe, R., *Correlation Properties of Cyclic Sequences*, JPL Technical Report No. 32-388, 1 July 1963, Jet Propulsion Laboratory, Pasadena, CA.

12. Watkins, J., Loftsson, J., and Tyler, S., A Binary Sequence of Period 60 with Better Autocorrelation Properties than the Barker Sequence of Periodic 13. *TDA Progress Report 42-82*, April-June 1985, Jet Propulsion Laboratory, Pasadena, CA.

13. Barlog, M., Horn, T., Ulrich, D., and Variot, M., *Binary String Manipulation*, CSUN Report CS480, May 1985.

Table 1. Best matched and mismatched sequences for periods 3 to 27

	Matched Sequence			Mismatched Sequence	
j	P	M	Sequence, hex	L	Sequence, hex
3	1	2	4	1.5000	4
4	0	0	E	1.0000	E
5	1	4	1E	1.1111	1E
6	2	20	25	1.3125	28
7	1	6	4B	1.5400	40
8	4	16	CB	1.3333	E5
9	3	24	1F4	1.6650	104
10	2	36	350	1.6761	25D
11	1	10	716	1.2909	67A
12	4	16	941	1.1250	941
13	1	12	1F35	1.0400	1E6B
14	2	52	36A3	1.2153	27F5
15	1	14	647A	1.1520	698F
16	4	48	FAC4	1.2589	EED8
17	3	64	19A3D	1.2165	1128E
18	2	68	31EDD	1.2843	21419
19	1	18	7A86C	1.1119	465D0
20	4	64	F6E8E	1.1111	C5640
21	3	52	117BCE	1.1097	170848
22	2	84	3D1231	1.2178	28312B
23	1	22	6650FA	1.1114	7CEA2D
24	4	32	DC20D4	1.0607	C3DEA6
25	3	72	18B082E	1.1195	128C0BC
26	2	100	2C1AEB1	1.1240	34AFBC9
27	3	74	5A3C444	1.0965	7D3472B

Table 2. Best matched and mismatched sequences for periods 28 to 64

	Matched Seqeunce			Mismatched Sequence		
j	*P*	*M*	Sequence, hex	*L*	Sequence, hex	Reference[a]
28	4	80	B30FDD4	1.1305	F3DDD21	9, 10
29	3	92	14A7C111	1.1384	14A7C111	9
30	2	116	3FAD938A	1.1260	3FAD938A	X
31	1	30	4B3E3750	1.0898	45C2D660	11, 10
32	4	80	89445BC1	1.0950	89445BC1	10
33	3	64	18A5C240D	1.0656	18A5C240D	9
34	2	132	29D3BB82D	1.1337	29D3BB82D	10
35	1	34	71F721592	(1.0732)	722B92F3E	11, X
36	4	64	F397A6517	1.0455	F397A6517	9
37	3	84	1BD623E3B6	1.0771	1BD623E3B6	9
38	2	148	302162B8B6	1.1085	302162B8B6	10
39	3	86	60CD4F47BE	1.0528	60CD4F47BE	X
40	4	80	DB9EAE05BC	1.0575	DB9EAE05BC	9
41	3	104	1079731045A	1.0723	1079731045A	10
42	2	164	2CF51397B7C	(1.0523)	2CF51397B7C	X
43	1	42	653BE2E08D6	(1.0786)	5189822FC34	11, X
44	4	144	AFDE8AF8665	1.1022	AFDE8AF8665	9
45	3	124	17473C9BFAD0	1.0667	17473C9BFAD0	9
46	2	180	2A2818CDBC16	(1.0842)	2A2818CDBC16	X
47	1	46	421A8D93A9EF	(1.0727)	795220A780EC	11, Y
48	(8)	96	99803C312AB6	1.0375	99803C312AB6	X
48	4	(112)	CBF089223A51			9
49	3	(144)	1C0B504676CB0	(1.0799)	1C0B504676CB0	X
50	2	196	236D4FF70651E	(1.0984)	236D4FF70651E	X
51	3	(146)	6D5DECF8433E8	(1.0704)	6D5DECF8433E8	X
52	4	(128)	FDEE871D85B44	1.0400	E8BEDC6F7AE24	X
53	3	(164)	11CAA3E46F7B65	(1.0706)	11CAA3E46F7B65	X
54	2	212	3917B588A2C302	1.0826	3917B588A2C302	X
55	3	(214)	7BCFB32717D0A5	(1.0837)	7BCFB32717D0A5	X
55	(5)	(182)	7F0AA13316DC34			X
56	4	(208)	852659EBA181B8	(1.0993)	852659EBA181B8	X
57	3	(184)	16A38C8BC7FD1AD	(1.0637)	16A38C8BC7FD1AD	X
58	2	228	3B64AAF8FDCE520	(1.0896)	3B64AAF8FDCE520	X
59	1	58	5D49DE7C1846D44	(1.1003)	6CF43BE8A12CF9D	11, Z
60	4	(112)	EC757781362D6F9	(1.0352)	EC757781362D6F9	12
61	3	(204)	1481F734DC7EEA74	(1.0624)	1481F734DC7EEA74	X
62	2	244	225746DC62583D20	1.0638	225746DC62583D20	9
63	1	62	4314F4725BB357E0	(1.0830)	408AB703D6597390	11, 13
64	4	(240)	B24FEAE7E4529CF0	(1.0538)	CD9BFF0E16D2AB98	ZZ, X

[a] X Refers to this article.
Y T. Safer, personal communication, April 1985.
Z N. Lee, personal communication, October 1984.
ZZ J. Watkins, personal communication, June 1985.